What is Claimed is:

1. A visual programming system, comprising:

one or more function modules each of which is provided with an applicable functional programming command stored in a computer executable language in a processing unit to accomplish a substantial applicable function; and

one or more commanding flow arrows connecting the function modules with each other in a predetermined sequence to construct a visual graphic program which is compiled to machine readable codes of said computer executable language, wherein information outputted from said applicable functional programming command of one of said function modules is inputted to another said applicable functional programming command of another said function module that is connected with said one of said function modules by said commanding flow arrow, so as to construct said visual graphic program in said computer executable language in said processing unit.

2. The system, as recited in claim 1, further comprising one or more determination modules each of which is provided with a determining test stored in computer executable language in the processing unit, wherein each of said determination modules is connected with three of said function modules and other said determination modules at an determination input, a "True" output and a "False" output thereof by three of said commanding flow arrows to construct said visual graphic program, wherein after said visual graphic program is compiled to machine readable codes, each of said determination modules conducts said determining test to send information inputted selectively to said function module or said determination module connected to said "True" output and said "False" output.

3. The system, as recited in claim 1, wherein each of said function modules has at least an information input for inputting information and an information output for outputting computed information, wherein when said commanding flow arrow has an arrow end connected to said information input of one of said function modules and another starting end connected to said information output of another said function module, information inputs to said applicable functional programming command of said function module from said another function module that outputs said information.

16

4.   The system, as recited in claim 2, wherein each of said function modules has at least an information input for inputting information and an information output for outputting computed information, wherein when said commanding flow arrow has an arrow end connected to said information input of one of said function modules and another starting end connected to said information output of another said function module, information inputs to said applicable functional programming command of said function module from said another function module that outputs said information.

5.   The system, as recited in claim 2, wherein each of said function modules has at least an information input for inputting information from one of said determination modules, when said commanding flow arrow has an arrow end connected to said information input of one of said function modules and another starting end connected to either said "True" output or said "False" output of said determination module, information inputs to said applicable functional programming command with respect to said function module from said determination module that outputs said information.

6.   The system, as recited in claim 1, wherein a construction of said function modules, said determination modules and said commanding flow arrows is displayed by said processing unit via a monitor thereof as said visual graphic program which directly represents said computer executable language to be stored in said processing unit to operate and function.

7.   The system, as recited in claim 4, wherein a construction of said function modules, said determination modules and said commanding flow arrows is displayed by said processing unit via a monitor thereof as said visual graphic program which directly represents said computer executable language to be stored in said processing unit to operate and function.

8.   The system, as recited in claim 5, wherein a construction of said function modules, said determination modules and said commanding flow arrows is displayed by said processing unit via a monitor thereof as said visual graphic program which directly represents said computer executable language to be stored in said processing unit to operate and function.

9. The system, as recited in claim 1, wherein human readable source codes of a source code program are converted and arranged into different said function modules of said visual graphic program according to a conversion rules database.

10. The system, as recited in claim 4, wherein human readable source codes of a source code program are converted and arranged into different said function modules and said determination modules of said visual graphic program according to a conversion rules database.

11. The system, as recited in claim 5, wherein human readable source codes of a source code program are converted and arranged into said different function modules and said determination modules of said visual graphic program according to a conversion rules database.

12. The system, as recited in claim 1, further including a user editing interface to construct said visual graphic program by selecting said function modules and linking said selected function modules by said commanding flow arrows, and a compiler which is used to convert said human readable source code program into said machine readable codes of said computer executable language following predetermined conversion instructions of a conversion rules database.

13. The system, as recited in claim 4, further including a user editing interface to construct said visual graphic program by selecting said function modules and said determination modules and linking said selected function modules and determination modules by said commanding flow arrows, and a compiler which is used to convert said human readable source code program into said machine readable codes of said computer executable language following predetermined conversion instructions of a conversion rules database.

14. The system, as recited in claim 5, further including a user editing interface to construct said visual graphic program by selecting said function modules and said determination modules and linking said selected function modules and determination modules by said commanding flow arrows, and a compiler which is used to convert said human readable source code program into said machine readable codes of said computer executable language following predetermined conversion instructions of a conversion rules database.

15. The system, as recited in claim 9, further including a user editing interface to construct said visual graphic program by selecting said function modules and linking said selected function modules by said commanding flow arrows, and a compiler which is used to convert said human readable source code program into said machine readable codes of said computer executable language following predetermined conversion instructions of said conversion rules database.

16. The system, as recited in claim 10, further including a user editing interface to construct said visual graphic program by selecting said function modules and said determination modules and linking said selected function modules and determination modules by said commanding flow arrows, and a compiler which is used to convert said human readable source code program into said machine readable codes of said computer executable language following predetermined conversion instructions of said conversion rules database.

17. The system, as recited in claim 11, further including a user editing interface to construct said visual graphic program by selecting said function modules and said determination modules and linking said selected function modules and determination modules by said commanding flow arrows, and a compiler which is used to convert said human readable source code program into said machine readable codes of said computer executable language following predetermined conversion instructions of said conversion rules database.

18. The system, as recited in claim 12, wherein said user editing interface comprises a command selection panel, a selected command panel, and a file management panel, wherein said command selection panel comprises selectable commands, including determining test commands in human readable programming languages, commanding flow arrows representing direction of flow of said program and functional commands, wherein when a command is selected from said command selection panel, said selected command appears in said selected command panel and, by arranging selected commands into a flow chart form, said visual graphic program is completed.

19. The system, as recited in claim 13, wherein said user editing interface comprises a command selection panel, a selected command panel, and a file management panel, wherein said command selection panel comprises selectable commands, including determining test commands in human readable programming languages, commanding

19

flow a rrows r epresenting d irection o f f low o f s aid p rogram a nd f unctional c ommands, wherein when a command is selected from said command selection panel, said selected command appears in said selected command panel and, by arranging selected commands into a flow chart form, said visual graphic program is completed.

20. The system, as recited in claim 14, wherein said user editing interface comprises a command selection panel, a selected command panel, and a file management panel, wherein said command selection panel comprises selectable commands, including determining test commands in human readable programming languages, commanding flow a rrows r epresenting d irection o f f low o f s aid p rogram a nd f unctional c ommands, wherein when a command is selected from said command selection panel, said selected command appears in said selected command panel and, by arranging selected commands into a flow chart form, said visual graphic program is completed.

21. The system, as recited in claim 15, wherein said user editing interface comprises a command selection panel, a selected command panel, and a file management panel, wherein said command selection panel comprises selectable commands, including determining test commands in human readable programming languages, commanding flow a rrows r epresenting d irection o f f low o f s aid p rogram a nd f unctional c ommands, wherein when a command is selected from said command selection panel, said selected command appears in said selected command panel and, by arranging selected commands into a flow chart form, said visual graphic program is completed.

22. The system, as recited in claim 16, wherein said user editing interface comprises a command selection panel, a selected command panel, and a file management panel, wherein said command selection panel comprises selectable commands, including determining test commands in human readable programming languages, commanding flow a rrows r epresenting d irection o f f low o f s aid p rogram a nd f unctional c ommands, wherein when a command is selected from said command selection panel, said selected command appears in said selected command panel and, by arranging selected commands into a flow chart form, said visual graphic program is completed.

23. The system, as recited in claim 17, wherein said user editing interface comprises a command selection panel, a selected command panel, and a file management panel, wherein said command selection panel comprises selectable commands, including determining test commands in human readable programming languages, commanding

flow arrows representing direction of flow of said program and functional commands, wherein when a command is selected from said command selection panel, said selected command appears in said selected command panel and, by arranging selected commands into a flow chart form, said visual graphic program is completed.

24. A visual programming method, comprising:

(a) assigning one or more function modules each of which is provided with an applicable functional programming command stored in a computer executable language in a processing unit to accomplish a substantial applicable function; and

(b) connecting said function modules in a predetermined sequence with one or more commanding flow arrows, each pointing from one direction to another to construct a visual graphic program; and

(c) compiling said visual graphic program to machine readable codes of said computer executable language by inputting information outputted from said applicable functional programming command of one of said function modules to another said applicable functional programming command of another said function module that is connected with said one of said function modules by said commanding flow arrow, so as to construct said visual graphic program in said computer executable language in said processing unit.

25. The method, as recited in claim 24, after the step (a), further comprising a step of assigning one or more determination modules each of which is provided with a determining test stored in said computer executable language in said processing unit, wherein each of said determination modules is connected with three of said function modules and other said determination modules at an determination input, a "True" output and a "False" output thereof by three of said commanding flow arrows to construct said visual graphic program, wherein after said visual graphic program is compiled to said machine readable codes, each of said determination modules conducts said determining test to send information inputted selectively to said function module or said determination module connected to said "True" output and said "False" output.

26. The method, as recited in claim 24, wherein each of said function modules has at least an information input for inputting information and an information output for

outputting computed information, wherein when said commanding flow arrow has an arrow end connected to said information input of one of said function modules and another starting end connected to said information output of another said function module, information inputs to said applicable functional programming command of said

5   function module from said another function module that outputs said information.

27.   The method, as recited in claim 2, wherein each of said function modules has at least an information input for inputting information and an information output for outputting computed information, wherein when said commanding flow arrow has an arrow end connected to said information input of one of said function modules and

10   another starting end connected to said information output of another said function module, information inputs to said applicable functional programming command of said function module from said another function module that outputs said information.

28.   The method, as recited in claim 25, wherein each of said function modules has at least an information input for inputting information from one of said determination

15   modules, when said commanding flow arrow has an arrow end connected to said information input of one of said function modules and another starting end connected to either said "True" output or said "False" output of said determination module, information inputs to said applicable functional programming command with respect to said function module from said determination module that outputs said information.

20   29.   The method, as recited in claim 25, wherein human readable source codes of a source code program are converted and arranged into different said function modules and said determination modules of said visual graphic program according to a conversion rules database.

30.   The method, as recited in claim 28, wherein human readable source codes

25   of a source code program are converted and arranged into different said function modules and said determination modules of said visual graphic program according to a conversion rules database.

31.   The method, as recited in claim 29, wherein said visual graphic program is constructed by a user editing interface by selecting said function modules and said

30   determination modules and linking said selected function modules and determination modules by said commanding flow arrows, and said human readable code program is

converted by a compiler into machine readable codes of said computer executable language following predetermined conversion instructions of said conversion rules database.

32. The method, as recited in claim 30, wherein said visual graphic program is constructed by a user editing interface by selecting said function modules and said determination modules and linking said selected function modules and determination modules by said commanding flow arrows, and said human readable code program is converted by a compiler into machine readable codes of said computer executable language following predetermined conversion instructions of said conversion rules database.

33. A method of allowing computer programs to be inputted without using advanced programming languages, comprising the steps of:

(a) establishing a conversion rule database containing conversion instructions of converting selectable commands to machine readable codes;

(b) providing a selection platform, wherein said selectable commands are listed out for a user to select a set of selected commands according to a desired flow of functions to be performed; and

(c) compiling said selected commands into machine readable codes according to said set of conversion instructions.

34. The method, as recited in claim 33, before the step (c), further comprising a sub-step of storing said selected commands inside a processing unit.

35. A method of allowing a designed computer program to be customized without using advanced programming languages, comprising the steps of:

(a) establishing a reverse conversion rule database containing reverse conversion instructions o f reverse converting machine readable codes of said designed computer program to human understandable codes;

(b) establishing a set of conversion rule database containing conversion instructions of converting selectable commands to machine readable codes;

(c) providing an imported code viewing platform, wherein said machine readable codes of said designed computer program are converted to and listed out as said human understandable codes according to said reversion conversion instructions;

(d) providing an editing platform, wherein selectable commands are listed out for a user to insert selected commands into said human understandable codes and deleting sections of said human understandable codes, forming a set of edited codes, according to a desired flow of functions to be performed; and

(e) compiling said edited codes into machine readable codes following said set of conversion rules.